REMARKS

This amendment is responsive to the Office Action dated March 18, 2005. Applicants have amended claims 1, 2, 5-9, 11, 20, 21, 23, 24, 33-37, 41, 44-47, and 50. In addition, Applicants have added claims 51-53. Claims 1-53 are pending upon entry of this amendment.

Claim Objections

In the Office Action, the Examiner objected to claims 23 and 41 as having incorrect dependencies. Applicants have amended claims 23 and 41 to correct the error.

Claim Rejection Under 35 U.S.C. § 112

In the Office Action, the Examiner rejected claims 21 under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which Applicants regards as the invention. In particular, the Examiner noted that step (g) was omitted. Applicants have amended claims 20 and 21 to remove the labels (a) – (j) for purposes of clarification. Applicants submit that claims, as amended, particularly point out and distinctly claim the subject matter, as required by 35 U.S.C. 112, second paragraph.

Claim Rejection Under 35 U.S.C. §§ 102, 103

Claims 1-19 and 33-50

In the Office Action, the Examiner rejected claims 1-8, 11 and 45-47 under 35 U.S.C. 102(e) as being anticipated by Jardin (USPN 6,681,327 B1). In addition, the Examiner rejected claims 9, 10, 12-19, 33-44, 48-50 under 35 U.S.C. 103(a) as being unpatentable over Jardin in view of Cohen et al. (USPN 6,389,462 B1), Maloney et al. (USPN 6,253,337 B1), Boeuf (USPN 6,009,502), Fujiyama et al. (USPN 6,052,728), Bellaton et al. (USPN 6,473,425 B1), Gelman et al. (USPN 6,415,329 B1), Holtey et al. (USPN 5,293,424) and Harper et al. (USPN 6,820,215 B2).

Applicants respectfully traverse the rejection to the extent such rejection may be considered applicable to the amended claims. The references fail to disclose each and every feature of the claimed invention, and provide no teaching that would have suggested the desirability of modification to include such features, either in singularly or in combination.

Applicants' amended claim 1 requires managing a communications negotiation between the client and the server through an intermediate device that supports both a direct mode and a proxy mode. Further, amended claim 1 requires decrypting encrypted data packets, and forwarding unencrypted data packets from the intermediate device to the server using a communication session negotiated by the client and the server when the intermediate device operates in direct mode. Amended claim 1 further requires forwarding unencrypted data packets from the intermediate device to the server using a second communication session negotiated by the server and the intermediate device when the intermediate device operates in proxy mode.

Applicants' amended claim 33 requires an acceleration apparatus adapted to operate in a direct mode. Claim 33 requires that in direct mode the acceleration apparatus decrypts data packets received from the client and forwards the decrypted data packets to one of the servers using a communication session negotiated by the client and the server, and in the proxy mode the acceleration apparatus responds to the client on behalf of the server and forwards the decrypted data packets to the server using a communication session negotiated by the acceleration device and the server.

Similarly amended claim 45 requires an SSL acceleration device having a communication engine that supports: (1) a direct mode in which decrypted data packets is forwarded to the servers using a communication session negotiated by the client and the server, and (2) a proxy mode in which the acceleration device responds to the client on behalf of the server and forwards the decrypted data packets to the server using the open communications session established by the acceleration device and the server.

For purposes of clarification, Applicants refer the Examiner to FIGS. 5 and 7 of the present application which illustrate a direct mode and a full proxy mode supported by the described intermediate acceleration device, respectively. The present application describes the direct mode, also referred to as a "cut through mode," as follows:

> Figure 5 illustrates a direct, cut through processing method. Packets from client to server are addressed from the client to the server and from server to client, with the intermediary, SSL device being transparent to both. In the embodiment shown therein, the SSL accelerator allows the client and server to negotiate the TCP/IP session directly, making only minor changes to the TCP/IP headers passing through the accelerator device, and tracking session data in a data structure in memory to enable SSL session handling to occur. As described herein, this mode is referred to herein as the "direct, cut-through"

> mode, since the client and server "think" they are communicating directly with each other, and the SSL accelerator is essentially transparent.

As illustrated in FIG. 5, client 100 and server 300 negotiate the TCP/IP session and operate as termination points for the session. In other words, in "cut through" (direct) mode, the acceleration device handles only the SSL session with the client, while the client and the server handle the TCP/IP session.

As described in further detail below with respect to Applicants' new claims, acceleration in direct (cut through) mode can lead to communication errors in certain situations due to the fact that the communication session spans from the client to the server.

In contrast to the direct mode, in "proxy mode" the acceleration device handles both the SSL session and the TCP communications with the client. As illustrated in FIG. 7, the acceleration device negotiates both the SSL session and the TCP session with the client device. The acceleration device negotiates a separate TCP communication session with the server. In this manner, the acceleration device operates as a termination point for the TCP communication session with the client, and communicates over an entirely separate TCP communication session with each of the servers.

As one example, the present application describes the full proxy mode as follows:

> Once the SSL and TCP sessions are established, the client can send SSL encrypted data to the accelerator 250. The SSL session is terminated on the accelerator 250 and decrypted SSL data is copied to the server's TCP session at step 270c. Likewise, after clear data is forwarded to the server and responded to (at step 275), clear data is received by the SSL accelerator at step 280, copied to the client's SSL session and returned in encrypted form to the client at step 280. The server's TCP session within the SSL device 250 is terminated on SSL device 250.

In contrast to the requirements of Applicants' claims, Jardin describe a broker that operates solely as a proxy on behalf of servers. At col. 4, ln. 36, Jardin specifically states that broker 100 operates as a proxy. Moreover, Jardin clearly describes broker 100 as a full proxy that negotiates the TCP/IP session on behalf of the servers:

> When communication between the client 110 and broker 120 **(which is a proxy for the intended recipient server 130)** is desired, the client 110 initiates a "handshake" as specified by the communication protocol, e.g., TCP/IP. For example, the client 110 transmits a packet having a set SYN bit from the client 110 to the broker 120. **The broker 120 responds to the client 110** by transmitting a packet having a set ACK bit, and the client 110 acknowledges the acknowledgement of the broker 120 by transmitting a packet having a set ACK bit. In addition to establishing a handshake pursuant to the

> network protocol, performing a handshake in accordance with the specification of the secure protocol (e.g., SSL) may be required.

With respect to communication between the broker and the server, Jardin describes rerouting the packets to the server using a different communication session and does not teach or suggest a second mode, i.e., a direct (cut through) mode. This is consistent with the fact that the Jardin broker acts as full proxy on behalf of the servers. For example, the Jardin broker may use a secure session to communicate with the servers. In this case "broker 120 initiates a secure SSL handshake with the server 130a (block 334) in a manner that is substantially similar to the handshake establishment between the client 110 and broker 120, described above." Alternatively, the Jardin broker may use an insecure session. In this case, "to reroute client packets to the server 130a, the broker 120 initiates a conventional (i.e., non-secure) handshake with the server 130a in accordance with the communication protocol (e.g., TCP/IP) specified between the broker 120 and server 130a (block 344), as described in the referenced application." Thus, in either case, the Jardin broker initiates a new communication session with the server.

For at least these reasons, Jardin in view of the other references fails to teach or suggest the requirements of independent claims 1, 33 and 45. None of the other references, either singularly or in combination, provide any teaching or suggests that overcome the deficiencies of Jardin.

For example, Cohen et al. describes a method for transparently redirecting an HTTP connection request that is directed from an origin server (107) to a proxy cache (110-1). Cohen makes clear that "[d]uring a handshaking procedure, a TCP connection is transparently established between the client (110-1) and the proxy cache."[1] Thus, like Jardin, Cohen describes a proxy architecture that uses separate TCP connections and fails to describe a direct or "cut through" mode.

Maloney et al. describes an information analysis system that is a combination of sensor, analysis, data conversion, and visualization programs. Boeuf describes a file server that stores data by allocating a single oversized contiguous storage area and by allowing data wrapping. Fujiyama et al. describes a network system in which each of multiple networks, each containing computers and relay computers, is connected to another network via multiple relay computers.

[1] Abstract.

None of the relay computers act as acceleration devices. Bellaton et al. (USPN 6,473,425 B1) describes a mechanism for dispatching a sequence of packets via a telecommunications network. Gelman et al. describes a method of communicating over a satellite or other high delay-bandwidth link that does not utilize TCP/IP. Holtey et al. describes a secure memory card and is unrelated to a network acceleration device. Harper et al. describes techniques for rejuvenating a component of a distributed data processing environment.

Consequently, none of the references, either singularly or in combination, teach or suggest managing a communications negotiation between the client and the server through an intermediate acceleration device that supports both a direct mode and a proxy mode, as required by amended claims 1, 33 and 45. None of the references teach or suggest modification of the Jardin broker to act as an intermediate acceleration device that operates in two modes: a proxy mode and a direct mode, as required by the amended independent claims.

Similarly, with respect to claim 2, none of the references teach or suggest receiving TCP session negotiation data from the client and modifying the negotiation data prior to forwarding the negotiation data to the server to establish the communications session between the client and the server when operating in direct mode. As described above, Jardin describes a proxy architecture in which the broker responds to the client and does describe an acceleration device that forwards negotiation data to a server to establish a communication session between the client and the server. For example, as quoted above, col. 4, ll.35-47, relied upon by the Examiner in rejecting claim 2, specifically states that "the client 110 transmits a packet having a set SYN bit from the client 110 to the broker 120. The broker 120 responds to the client 110 by transmitting a packet having a set ACK bit." Thus, the Examiner appears to have overlooked that the Jardin broker does not modify the negotiation data at all, and does not forward the negotiation data to the server to establish the communications session between the client and the server. To the contrary, the Jardin broker responds to the client on behalf of the server.

With respect to claims 3 and 4, Jardin in view of the cited references fails to describe modifying a SYN request or a maximum segment size and then forwarding the modified negotiation data to the server. As noted above, col. 4, ll.35-47 of Jardin, relied upon by the Examiner in rejecting claim 2, specifically states that the broker 120 responds to the client 110 and does not modify the SYN request and forward the negotiation to the server. In fact, the Jardin broker does not modify the SYN request at all.

For at least these reasons, withdrawal of the rejections of claims 1-19 and 33-45 under 35 U.S.C. § 102 and 103 is requested.

Claims 20-30

The Examiner rejected claims 20-30 under 35 U.S.C. 103(a) as being unpatentable over Bellwood et al. (USPN 6,584,567 B1) in view of Maloney et al. (USPN 6,253,337 B1), Cohen et al., Bellaton et al., Holtey et al., Boeuf (USPN 6,009,502), and Weinstein et al. (USPN 6,094,485).

Applicants' amended claim 20 recites steps performed by an intermediate device when using the "direct" (cut through) communication session described above. For example, claim 20 requires establishing a communications session between the client and said one of said plurality of servers by receiving negotiation data from the client intended for the server and forwarding the negotiation data in modified form to the server, and receiving negotiation data from the server intended for the client and forwarding the negotiation data to the client to establish the client and the server as terminations for the communications session. In this manner, a communication session is established through the intermediate device and terminated by client and the server.

In addition, claim 20 requires established establishing a secure communications session between the client and the intermediary device, and forwarding decrypted application data from the intermediary device to said one of said plurality of servers using the communications session established between the client and the server.

In contrast, Bellwood describes a method of enabling a **proxy** to participate in a secure communication between a client and a set of servers. The method begins by establishing a first secure session between the **client and the proxy**. Upon verifying the first secure session, the method continues by establishing a second secure session between the **client and the proxy**.[2]

Thus, Bellwood is not describing a direct acceleration method a communication session is established through the intermediate acceleration device and terminated by client and the server, where the intermediate device establishing a secure communications session between the client

[2] Abstract.

and the intermediary device, and forwards decrypted application data from the intermediary device to said one of said plurality of servers using the communications session established between the client and the server, as required by claim 20.

As described above, none of the other references, either singularly or in combination, provide any teaching or suggestion that overcomes the deficiencies of Bellwood. Cohen describes a proxy architecture that uses separate TCP connections and fails to describe a direct or cut through mode. Maloney et al. describes an information analysis system that is a combination of sensor, analysis, data conversion, and visualization programs. Boeuf describes a file server that stores data by allocating a single oversized contiguous storage area and by allowing data wrapping. Fujiyama et al. describes a network system in which each of multiple networks, each containing computers and relay computers, is connected to another network via multiple relay computers. None of the relay computers act as acceleration devices. Bellaton et al. describes a mechanism for dispatching a sequence of packets via a telecommunications network. Gelman et al. describes a method of communicating over a satellite or other high delay-bandwidth link that does not utilize TCP/IP. Holtey et al. describes a secure memory card an is unrelated to a network acceleration device. Harper et al. describes techniques for rejuvenating a component of a distributed data processing environment.

For at least these reasons, withdrawal of the rejections of claims 20-33 under 35 U.S.C. § 102 and 103 is requested.

New Claims:

Applicants have added claims 51-53 to the pending application. The applied references fail to disclose or suggest the inventions defined by Applicants' new claims, and provide no teaching that would have suggested the desirability of modification to arrive at the claimed inventions.

As one example, the references fail to disclose or suggest automatically switching the intermediate device from the direct mode to the proxy mode upon detecting a communication error associated with the direct mode, as recited by claim 51. As another example, the references fail to disclose or suggest an acceleration apparatus that automatically switches from the direct mode to the proxy mode upon detection of a communication error associated with the communication session negotiated by the client and the server, as required by claim 52.

No new matter has been added by the new claims. For example, on pages 25-26, the present application describes switching to a proxy mode from a cut through mode upon detecting communication problems:

> Figure 9b shows a further feature of the device, allowing for mode switching: the system can begin a full TCP proxy mode session (in accordance with the description of Figure 6) and switch to cut through/direct modes depending on the circumstances of the data transfer. Full proxy TCP mode has the advantage that all cases of transmission are supported. However, this embodiment requires more buffer memory than TCP cut through mode shown in Figure 5.
>
> In the cut through modes, certain types of packet transmissions can cause problems. For example, when the SSL record transverses more than one TCP segment or when the client window is very small, (for example, on the order of 200 – 300 bytes) and many small TCP segments are received.
>
> The switching mode shown in Figure 9b can therefore allow the TCP proxy mode for SSL and TCP session setup, and then cut through mode for normal data, with a roll back to the proxy TCP mode for problem cases.

CONCLUSION

All claims in this application are in condition for allowance. Applicants respectfully request reconsideration and prompt allowance of all pending claims. Please charge any additional fees or credit any overpayment to deposit account number 50-1778. The Examiner is invited to telephone the below-signed attorney to discuss this application.

Date:

June 20, 2005

SHUMAKER & SIEFFERT, P.A.
8425 Seasons Parkway, Suite 105
St. Paul, Minnesota 55125
Telephone: 651.735.1100
Facsimile: 651.735.1102

By:

Name: Kent J. Sieffert
Reg. No.: 41,312